

## CLAIMS

- 1 1. A method for multiplexing data in a packet network system with  
2 a plurality of data queues, said method comprising the steps:
  - 3 a) initialize a first output queue section by making all  
4 address pointers in that section point to their own  
5 addresses;
  - 6 b) initialize said system by setting an input queue equal to  
7 a first input stream and initialize a next section of its  
8 output queue by making all address pointers in that output  
9 queue point to their own addresses;
  - 10 c) select the next input queue;
  - 11 d) read a next packet from said next input queue and select a  
12 location for it in said output queue;
  - 13 e) determine whether said selected location in said output  
14 queue is occupied by examining its address pointer;
  - 15 f) if said selected location is occupied, go to the address  
16 pointed to by the address pointer for said selected  
17 location;
  - 18 g) if said location pointed to by the address pointer is not  
19 occupied, store the pointer to a second input stream in  
20 said selected location;
  - 21 h) modify all output queue addresses accessed by steps d and  
22 f to point to the next unoccupied location in the output  
23 queue;
  - 24 i) go to step c and repeat until the last input queue in said  
25 plurality of data queues is multiplexed.
- 1 2. The method of Claim 1 wherein the packet network is an Internet  
2 Protocol network.

1 3. The method of Claim 2 wherein the data is MPEG-2 transport  
2 stream data.

1 4. The method of Claim 2 wherein the data is digital uncompressed  
2 stream data.

1 5. The method of Claim 2 wherein the data is digital compressed  
2 stream data.

1 6. A method for multiplexing data in a packet network system with  
2 a plurality of data queues, said method comprising the steps:  
3 a) initialize a first output queue section by making all  
4 address pointers in that section point to their own  
5 addresses;  
6 b) initialize said system by setting an input queue equal to  
7 a first input stream and initialize a next section of its  
8 output queue by making all address pointers in that output  
9 queue point to their own addresses;  
10 c) select the next input queue;  
11 d) read a next packet from said next input queue and select a  
12 location for it in said output queue;  
13 e) determine whether said selected location in said output  
14 queue is occupied by examining its address pointer;  
15 f) if said selected location is occupied, go to the address

16 pointed to by the address pointer for said selected  
 17 location;

18 g) if said location pointed to by the address pointer is not  
 19 occupied, store the pointer to a second input stream in  
 20 said location;

21 h) modify all output queue addresses accessed by steps d and  
 22 f to point to the next unoccupied location in the output  
 23 queue;

24 i) determining whether this is the last input queue;

25 j) if this is the last input queue, send out the output  
 26 queue;

27 k) if this is not the last input queue, determine whether  
 28 this is the last packet in the selected input queue;

29 l) if this is the last packet in the selected input queue, go  
 30 to step c; and

31 m) if this is not the last packet in the selected input  
 32 queue, go to step d.

1 7. The method of Claim 6 wherein the packet network is an Internet  
 2 Protocol network.

1 8. The method of Claim 7 wherein the data is MPEG-2 transport  
 2 stream data.

1 9. The method of Claim 7 wherein the data is digital uncompressed  
 2 stream data.

